REMARKS

Claims 1-34 are presented for examination. Claims 1, 3, 12, 14, 23, and 25 are presently amended. No claims are canceled.

Claims 1-34 were rejected under 35 U.S.C 102(b) as being anticipated by Nagata et al., U.S. Pat. 5,058,035. Applicants respectfully disagree.

Amendments to claims 3, 14, and 25 address idiomatic or antecedent oversights.

Claims 1, 12, and 23 are amended to more clearly recite claim limitations and to remove any unintended ambiguity. The Office Action appears to have equated the nonvolatile memory recited in the present claims with the nonvolatile RAM recited in Nagata et al. The two memories have architecturally different uses.

The nonvolatile RAM of Nagata et al. is used as a second command RAM to hold executable commands (Col. 4, lines 7-10). As it known in art, a computing device typically has a ROM to hold a boot-up sequence, and the contents of the ROM are transferred to a command register, or RAM, for execution. Nagata et al. explain that if no data is stored in their nonvolatile RAM (i.e. initial-state buffer 205), then their printer will follow this known boot-up sequence. However, if an alternate boot-up sequence is stored in the nonvolatile RAM, then it is executed in place of the typical boot-up sequence (Col. 4, line 62 to Col. 5, line 2). As the boot-up sequence is executed, initialization commands are executed (Col. 5, lines 2-6) to generate parameter settings data that are written to various parameter registers, or ports, which together may constitute a second RAM to hold these processed parameter values (Col. 3, lines 42-45). The commands that are executed to set parameter setting values may also be transmitted by the host device one-at-a-time (Col. 3, lines 38-39). Thus, Nagata et al.'s nonvolatile RAM holds executable commands (just like executable commands sent from the host) that, when executed (i.e. processed) write parameter setting data to ports (i.e. registers), which as group constitute a

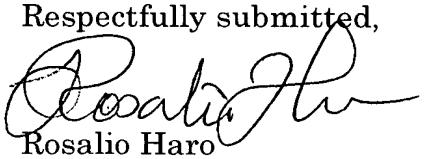
second RAM for holding the processed parameter settings data (i.e. processed parameter values).

By contrast, the nonvolatile memory recited in the present claims does not hold executable commands. Rather, it receives the parameter setting values resulting from the processing of executable parameter setting commands sent from the host. To remove any unintended ambiguity, claims 1, 12, and 23 are amended to more clearly recite that the volatile memory stores "operating parameter values obtained from the processing of specific one or more commands from said host device". This is in direct conflict with the teachings of Nagata et al., which require that the commands in his initial state buffer be executed one-at-a-time to re-process parameter setting commands and re-write the processed parameter values one-at-a-time to the appropriate ports, or registers. By contrast in the present invention, it is not necessary to re-execute the parameter setting commands each time the volatile memory of the present invention is to be updated since the previously processed parameter values are saved to, and restored from, a nonvolatile memory.

It is respectfully submitted that such a feature is neither taught or suggested in the cited prior art. Each of the remaining claims depend directly or indirectly on independent claims 1, 12 and 23, and are thus patentable for at least the same reasons as discussed above.

In view of the foregoing amendments and remarks, Applicants respectfully request favorable reconsideration of the present application.

Respectfully submitted,



Rosalio Haro
Registration No. 42,633

Please address all correspondence to:

Epson Research and Development, Inc.
Intellectual Property Department
150 River Oaks Parkway, Suite 225
San Jose, CA 95134
Phone: (408) 952-6000
Facsimile: (408) 954-9058
Customer No. 20178

Date: April 7, 2004